3К

УДК 004.827

Ю.В. Крак, Ю.В. Барчукова, Б.А. Троценко
Інститут кібернетики імені В.М. Глушкова НАН України, м. Київ, Україна
Yuri.Krak@gmail.com

Побудова моделей дактилем для синтезу дактильної інформації

У статті розглядається підхід до побудови специфікацій рухів людини, а саме дактильної жестової мови. Розглядається інформаційна модель руки людини. Описано процес побудови специфікації для дактильної жестової мови. Проведено порівняльний аналіз семи жестових мов. Запропоновано підхід та програмну реалізацію на платформі .Net системи передачі асинхронних повідомлень, що застосовуються для проведення паралельних обчислень на усіх ядрах процесора, уникнення блокування основного потоку програми, написання сценаріїв діяльності програми.

Вступ

Розвиток комп'ютерної техніки, виникнення та розвиток новітніх підходів до аналізу, організації, зберігання та подання інформації робить можливим створення інформаційних технологій у сфері моделювання та аналізу комунікаційної жестової мови [1]. Перспективними з точки зору практичних застосувань були виділені напрямки розробок зі створення систем навчання українській жестовій мові (УЖМ) з використанням тривимірних моделей людини, а саме: розробка інформаційно-навчального комплексу «Українська дактильна абетка».

Особливі потреби глухих людей спонукають до більш детального вивчення процесу побудови жестової мови та синтезу елементів жестової мови. В даній роботі розглядається процес побудови дактильної абетки української жестової мови. Задача полягає в тому, щоб на основі текстового опису дактилеми отримати повну інформацію про жест, який необхідно побудувати на тривимірній моделі руки. Складність полягає в тому, що текстовий опис, який наразі існує для опису дактилем української дактильної абетки, був розроблений для навчального процесу [2], фотографічні знімки та відео не передають структуру дактилеми з кутом огляду в 360°, а отже, виникає складність у визначенні положень пальців в закритих позиціях. Необхідно створити такий опис дактилем, який би дозволив однозначно визначати положення пальців руки. Іншою проблемою є те, що кожна людина-носій жестової мови відповідно до її унікальності та початкового джерела навчання жестовій мові вивчає та засвоює мову не однозначним чином, а додає свій «діалект» до жестів. А звідси виникає проблема визначеності в «правильній» вимові. Тому існує необхідність в створенні такої моделі руки, яка б відповідала встановленим нормам до показу дактилем. При побудові інформаційної моделі руки людини необхідно враховувати ці особливості і водночас нехтувати ознаками, які є несуттєвими при показі дактилем (наприклад, рівень зігнутості пальця може бути різним в кожної людини при показі однієї і тієї ж дактилеми, але це не говорить про те, що існує така кількість варіантів показу однієї і тієї ж дактилеми). Отже, задача полягає в тому, щоб на основі формального текстового опису дактилем отримати його комп'ютерне представлення у вигляді тривимірної моделі руки.

Метою даної роботи є аналіз побудови української дактильної абетки та формалізація опису дактилем для задачі моделювання рухів людини на тривимірній моделі людини та синтезу дактильної інформації; розробити комп’ютерну технологію на основі проведених досліджень.

Аналіз останніх досліджень і публікацій

Опис людських рухів – важка задача з великим відсотком невизначеності відповідно до людського тіла. Проблемі вивчення жестикуляції з лінгвістичної точки зору присвячені праці таких вчених, як У. Стоку (АЖМ), Г.Л. Зайцевої (РЖМ) та ін. У світі існують також системи специфікації жестової мови. Одна з таких систем – Гамбурзька система Hamburg Notation System (HamNoSys) [3]. Вона включає в себе різноманітні графічні позначення, завдяки чому можливо описати великий спектр жестів. Також широкого застосування набула система SignWriter (Valerie Sutton) [4]. У цих системах використовуються графічні позначення в комбінації з текстовими позначеннями, що в термінах комп’ютерного моделювання не завжди зручно для використання, оскільки для моделювання необхідно мати інформацію в термінах комп’ютерного представлення.

У даній роботі ставиться задача дослідити процес побудови жестів української дактильної абетки з точки зору формалізації рухів для побудови формальної моделі руки для задач моделювання тривимірної моделі руки, синтезу та розпізнавання жестів.

Аналіз та формалізація дактильної абетки на основі природних характеристик

Параметри побудови дактилеми

У. Стоку виділив три параметри для опису структури жесту: положення жесту, форма руки (її конфігурація та орієнтація) та рух [5]. Базуючись на даній класифікації, виділимо три основні параметри, за якими відбувається аналіз дактилем:

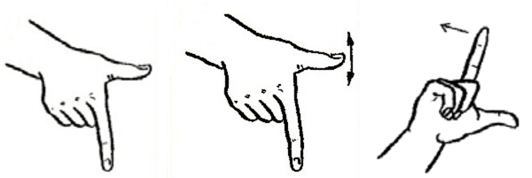
1. Конфігурація пальців руки.
2. Орієнтація руки в просторі.
3. Характер руху.

У даній роботі особливу увагу приділено дослідженню саме конфігурації кисті руки, оскільки в процесі побудови тривимірної моделі руки людини головним є визначення правильної конфігурації пальців руки, а вже потім розташування руки в просторі та анімація руху. На даному етапі положення руки в просторі відносно промовця не вразовується; вважається, що рука – це об’єкт у просторі. Для побудови специфікацій дактильних жестових одиниць (дактилем) використано наступні параметри: конфігурація пальців, орієнтація руки, рух руки.

Конфігурація пальців – це визначення таких параметрів, як перелік пальців, що задіяні в побудові дактилеми, конфігурація пальців (рівень нахилу, зігнутості) та взаємне розташування відносно один одного.

Українська дактильна абетка складається з 33 дактилем, з яких 24 мають різну конфігурацію пальців (А, Б, В, Г, Е, Є, Ж, З, И, І, Л, М, Н, О, П, Р, С, Т, У, Ф, Х, Ч, Ю, Я), решта 9 мають аналогічну конфігурацію пальців та відрізняються від попередніх орієнтацією руки (наприклад Ш) та (або) рухом (наприклад Ь, Ї, К).

Таблиця 1 – Дактилеми з однією конфігурацією пальців та різною орієнтацією руки в просторі та рухом

Дактилеми	Зображення
Г – Ґ – Ь	

Інформаційна модель дактилеми

Інформаційна модель дактилеми – це впорядкована трійка *{K, O, M},* де параметр *K* описує конфігурацію пальців, *O* – орієнтацію руки в просторі, *M* – рух.

Параметр *K* – впорядкована шестірка $\{f_1, f_2, f_3, f_4, f_5, c\}$, де параметри f_1, f_2, f_3, f_4, f_5 описують конфігурацію кожного пальця окремо, *с* – взаємне розташування між пальцями (вказівний, середній, безіменний, мізинець), що використовуються для побудови дактилеми.

f_1 – конфігурація великого пальця,
f_2 – конфігурація вказівного пальця,
f_3 – конфігурація середнього пальця,
f_4 – конфігурація безіменного пальця,
f_5 – конфігурація мізинця.

Значення параметра f_1 наведено в табл. 2.

Таблиця 2 – Значення параметра f_1

Значення	Опис	Приклад
1	Рівно	А (Українське)
2-α	Відхиляється в площині долоні на α градусів	Г (Українське)
3-1	Поверх кулака (або покриває декілька пальців)	Я, І (Українське)
3-2	Поверх вказівного пальця	
3-3	Поверх середнього пальця	Р (Українське)
3-4	Поверх безіменного пальця	Н (Українське)
3-5	Поверх мізинця	Т (Українське)
4	Всередині кулака (на долоні)	В, Х (Українське)
5-0	Утворює дугу з іншими пальцями не замкнену	С (Українське)
5-1	Утворює дугу з іншими пальцями замкнену	О (Українське)
5-2	Утворює перехрест з пальцем	Т (Польське)
6	Горизонтально, підпирає пальці	Е (Американське)
7-3	Подушечкою спирається на середній палець	Т (Американське)
7-4	Подушечкою спирається на безіменний палець	N (Американське)
7-5	Подушечкою спирається на мізинець	М (Американське)
8	Паралельно площині долоні	К (Американське)

Параметр $f_i = x \ [y]$, $i = \{\overline{2,5}\}$, де $x \in [0,1]$, $y \in [0,1]]$. x має зміст – рівень нахилу пальця *i* відносно рівного стану ($x = 1$), y – рівень зігнутості пальця *i* (параметр y може бути відсутній в описі, якщо описання зігнутості не потребується).

Рівний стан (x=1) – еквівалентний конфігурації руки при показі дактилеми «В» (всі пальці випрямлені та знаходяться на площині, що утворюється долонею).

Закритий стан (x=0) еквівалентний конфігурації руки при показі дактилеми «А» (пальці зігнуті в кулак).

Параметр c визначається наступним чином:

$$c = \begin{cases} 0, & \text{якщо з 4х пальців в анімації задіяно 1 або жодного} \\ 1, & \text{якщо пальці піднято рівно вгору або притиснуто один до одного} \\ 2, & \text{якщо пальці не дотикаються один до одного} \\ 3, & \text{якщо пальці заходяться один поверх іншого (дактилема «Я»)} \\ 4, & \text{якщо пальці зміщуються в різні площини (дактилема «Б»)} \end{cases}$$

Приклад визначення специфікацій для конфігурації пальців руки наведено в табл. 3.

Таблиця 3 – Специфікація конфігурації пальців руки

Дактилема	f_1	f_2	f_3	f_4	f_5	c	Зображення
Х	4	0	0	0	0,2 [0,2]	0	
К	3-1	0	0	1	1	2	
Е	5-1	0,5 [0,5]	0,5 [0,5]	0,5 [0,5]	0,5 [0,5]	1	
Я	3-1	0	0	1	1	3	

Параметр O – впорядкована трійка {α, β, γ}, де α, β, γ – кути повороту руки в просторі відносно початкового стану.

Під початковим станом розуміється орієнтація руки *O {0,0,0}* . При дактилюванні дактилеми «В», рука повертається на кут α = 180°, тобто долонею до співбесідника.

Параметр M описує характер руху. На даному етапі побудови специфікації параметр руху описується наступним чином:

$$M = \begin{cases} 0, & \text{рух відсутній} \\ 1, & \text{рух здійснюється частиною руки (наприклад «Ґ»)} \\ 2, & \text{рух описовий (наприклад «З», «Д»)} \\ 3, & \text{зміщення в просторі, поворот (наприклад «К», «Щ»)} \\ 4, & \text{перехід від однієї конфігурації пальців в іншу} \end{cases}$$

На основі визначених параметрів була побудована таблиця специфікацій з 12 параметрами, що наведено в табл. 4.

Таблиця 4 – Таблиця специфікацій дактилем

Мова	Дактилема	f_5	f_4	f_3	f_2	f_1	c	α	β	γ	M
Укр	У	1	0	0	0	2-45	2	180	0	0	0

Дані позначення є легкими для сприйняття в контексті моделювання рухів та інформативні. Завдяки ним легко будувати специфікації для інших дактильних абеток.

Аналіз дактилем

При побудові специфікацій дактилем для української дактильної абетки також було проведено дослідження наступних дактильних абеток: російської, польської, грецької, американської, німецької та французької. Аналіз специфіки побудови жестів руки в різних мовах надає змогу побудувати таку специфікацію, що дозволяє описувати дактилеми не тільки УЖМ, а й інших одноручних дактильних абеток.

На основі аналізу УДЖМ було проведено аналіз інших мов та побудована таблиця специфікацій для української, американської, німецької, французької, польської, російської та грецької дактильної абетки (табл. 5).

Таблиця 5 – Результати дослідження дактильних абеток

Дактильна абетка	Українська	Американська	Німецька	Французька	Польська	Російська	Грецька
Кількість дактилем	33	26	30	26	32	33	24
Кількість конфігурацій	24	23	24	21	19	24	21

За отриманими специфікаціями було проведено порівняльний аналіз української дактильної абетки з зазначеними раніше мовами. Було порівняно визначені раніше параметри конфігурації пальців руки, орієнтацією руки в просторі та рухом було знехтувано, оскільки будь-яка дактилема утворюється шляхом побудови конфігурації пальців та визначенням певної орієнтації руки в просторі та додаванням руху. Результати аналізу наведені в табл. 6. У таблиці наведена кількість однакових дактилем відповідно до визначених специфікацій.

1 – кількість існуючих конфігурацій в даній мові.

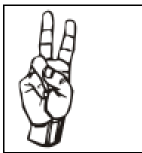
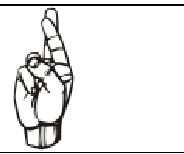
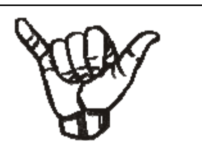
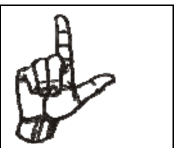
2 – кількість конфігурацій, що збігаються із 24 конфігураціями української дактильної абетки.

Таблиця 6 – Результати порівняння конфігурації пальців дактильних абеток

	Амер-ка	Польська	Грецька	Російська	Німецька	Фран-ка
1	23	19	21	24	24	21
2	12	11	10	21	13	10

Також було проведено порівняльний аналіз кожної з цих дактильних абеток між собою. Можна визначити, що 5 конфігурацій є спільними для усіх 7 мов.

Таблиця 7 – Конфігурації пальців, що є спільними для 7 мов

За побудованими специфікаціями було проведено аналіз 7 зазначених дактильних абеток, а саме конфігурацій пальців. Із 156 конфігурацій пальців, що визначають 204 дактилеми сімох жестових мов, унікальними є 50 конфігурацій. Позначимо цю множину за *U*. Звідси випливає, що будь-яку дактилему з цих семи дактильних абеток можна отримати із множини *U* шляхом змінення орієнтації руки в просторі та (або) додаванням руху. Можна зробити висновок, що можливо побудувати множину моделей руки людини, що буде однозначно визначати конфігурацію пальців руки людини для дактильних абеток української, російської, польської, грецької, американської, німецької та французької жестових мов.

Система передачі асинхронних повідомлень як засіб організації паралельних обчислень і її використання для задачі моделювання жестової мови

Традиційний підхід до створення програмного забезпечення орієнтується на використання послідовних алгоритмів. Цей вибір був очевидним, оскільки десятиліттями обчислювальні машини в основному використовували один процесор. Окрему увагу слід приділити переходу до сучасної парадигми подій як засобу комунікації між операційною системою та програмою: типова програма має принаймні один потік, що працює у циклі обробки повідомлень від операційної системи, що інформують програму про ту чи іншу подію [6]. Як наслідок, основні середовища та інструментальні засоби розробки програмного забезпечення (Visual Studio, XCode, .Net, Cocoa, QT, java) за замовчуванням орієнтуються саме на однопоточну парадигму обробки подій. Ці фактори значно вплинули на якість більшості програм, які використовуються в сьогоденні.

Слід виділити дві проблеми, пов’язані з даним підходом (парадигмою):

– «Підвисання» інтерфейсу користувача при проведенні обчислень або операцій вводу/виводу.

– Неповне використання апаратних ресурсів комп’ютера при проведенні обчислень, які мають альтернативні паралельні алгоритми.

Перевагою є підтримка зі сторони інструментальних засобів розробки, як наслідок, відносна простота підходу.

Для багатьох програмних середовищ сформовано шаблони, набори правил та (або) бібліотеки, що певною мірою вирішують ці недоліки. Серед основних слід виділити бібліотеки Grand Center Dispatch (програмна бібліотека асинхронних повідомлень для середовища Objective C, надбудова над Cocoa), Task Parallel Library, Rx Framework; шаблони DoWorkAsync/OnWorkCompleted, BeginWork/EndWork.

Пропонується альтернативний підхід та програмна реалізація у вигляді бібліотеки для платформи .Net для використання потоків обробки даних у пріоритетній (зваженій) черзі та передача асинхронних повідомлень (задач) як засіб вирішення зазначених проблем.

Теоретична частина

На нижчому рівні абстрагування для розвантаження основного потоку обробки подій та виконання обчислень (завдань) на різних ядрах процесора(ів) використовуються додаткові потоки, робота (задачі) яким передаються за допомогою повідомлень. Як і у стандартному підході до обробки подій від операційної системи, кожен з таких потоків знаходиться в однотипному циклі обробки повідомлень:

```
поки (потрібно_продовжувати_роботу):
п = отримати_перше_повідомлення_з_черги
обробка_повідомлення(п)
```

Доступ до черги можливий з різних потоків та синхронізується за допомогою мютексів. Відповідно, під час виконання процедури обробка_повідомлення можливе додавання повідомлень (задач) у чергу.

На вищому рівні абстрагування робота поділяється на типи, визначені користувачем, та виконується у класах-абонентах, що підписані на виконання роботи різного типу. Кількість фактичних потоків, що проводитимуть виконання, абстрагована; у граничному випадку таких може і не бути. Черга повідомлень є частиною класу-абонента.

Центральним місцем обміну повідомленнями є окремий клас – шина, на якому також повинні бути зареєстровані усі класи-абоненти і який є проміжною ланкою при передачі повідомлень. Шина також виконує контролюючу функцію та зберігає лічильник кількості повідомлень, що знаходяться в обробці.

Поза типовим сценарієм функціонування програми, слід окремо розглянути завершення програми. Для цього передбачається окремий тип повідомлення. Таке повідомлення дозволяється передати через шину лише 1 раз; на відміну від решти повідомлень, повідомлення про завершення не передається відразу до класу-абонента, а зберігається на шині, поки лічильник кількості повідомлень, що знаходяться в обробці, не стане рівним нулю. Тоді повідомлення про завершення буде передано відповідному класові-абоненту – таким чином спроектовано сценарій завершення програми. Отже, центральними функціональними класами є шина, повідомлення та абонент.

Практичний опис

MessagingBus – імплементація класу «шина». Програма повинна створити екземпляр цього класу для роботи з бібліотекою асинхронних повідомлень.

GeneralSubscriber – імплементація «класу-абонента». Кожен клас, що реєструватиметься на шині, повинен наслідувати даний. У ньому:

– Id – властивість, ідентифікаційний номер абонента,

– WorkerThreadPriority – властивість, пріоритет потоків виконання роботи,

– ProcessMessage(msg) – abstract – процедура обробка_повідомлення.

GeneralSingleThreadedSubscriber – клас-абонент, що використовує один потік обробки.

GeneralMultiThreadedSubscriber – клас-абонент, що використовує задану кількість потоків обробки. У ньому:

– numberOfWorkingTheads – властивість, кількість потоків обробки.

GeneralThreadlessSubscriber – клас-абонент, що для кожного повідомлення використовує окремо створений потік. GeneralDispatcherSubscriber – клас-абонент, що використовує основний потік для виконання роботи.

GeneralMessage – імплементація класу «повідомлення». У ньому:

– Id – властивість, ідентифікаційний номер повідомлення,

– TargetSubscriberId – властивість, ідентифікаційний номер абонента для даного повідомлення,

– Priority – властивість, пріоритет повідомлення,

– ErrorHandler – властивість для вказання обробника помилок,

– OnMessageProcessed – подія, що оголошується після закінчення обробки повідомлення.

SequentialMessage, наслідник GeneralMessage, приймає як параметр масив інших повідомлень, які по черзі передаються на шину обміну; кожне повідомлення передається після завершення обробки попереднього. Власне SequentialMessage оголошується завершеним після завершення останнього повідомлення з масиву.

ParallelMessage, наслідник GeneralMessage, приймає як параметр масив інших повідомлень, які відразу передаються на шину обміну. Власне ParallelMessage оголошується завершеним тоді, коли усі повідомлення з масиву завершені.

WaitMessage, наслідник GeneralMessage, приймає як параметр інтервал часу t; обробка WaitMessage на шині займає t часу. Використовується для створення планованих пауз.

TimeCheckedMessage, наслідник GeneralMessage, приймає як параметр інтервал часу t, повідомлення controlledMessage, повідомлення longExecution, повідомлення longExecutionFallback. Обробка даного повідомлення полягає в передачі на шину повідомлення controlledMessage; якщо його обробка займає більше, ніж t часу, на шину також передається повідомлення longExecution – у такому випадку також буде передано повідомлення longExecutionFallback, але після завершення обробки повідомлення controlledMessage. Даний тип повідомлень (TimeCheckedMessage), наприклад, використовується для сценаріїв, коли основне повідомлення типово займає невеликий проміжок часу, але потрібно передбачити повідомлення для користувача про діяльність програми у разі довгої обробки основного повідомлення.

Характеристика

Шина приймає повідомлення та асинхронно додає їх у чергу того класу-абонента, що відповідає полю TargetSubscriberId повідомлення. Це дозволяє розробляти так звані «слабозалежні» системи за рахунок підміни класу-абонента, який буде використовуватись у кінцевій версії програмного продукту. Як наслідок, легше створювати і тестувати окремі модулі програм; легше організовувати взаємодію кількох програмістів у команді.

Висновки

Дані дослідження є важливим етапом на шляху побудови якісної тривимірної моделі руки людини. Отримана інформаційна модель дактилем дозволяє ефективно будувати специфікації для дактильних абеток різних жестових мов. Дослідження української дактильної абетки в рамках цієї роботи сприяє розвитку комп’ютерних технологій для вивчення цієї дактильної жестової мови, а побудова специфікацій дактильних абеток різних жестових мов та їх порівняльний аналіз є підґрунтям для розробок систем навчання дактильній абетці різних мов. Підхід передачі асинхронних повідомлень показав свою ефективність на практиці при створенні програмних засобів «Українська жестова мова» та «Українська дактильна жестова мова».

Подальша робота буде направлена на дослідження руху при побудові дактилем; підвищення абстрактизації бібліотеки, зокрема, виділення черг роботи, що дозволить створювати різнотипні класи-абоненти. Це, для прикладу, дозволить проводити обчислення окремо на центральному і на графічному процесорах, делегувати обчислення на інші комп’ютери в мережі.

Література

1. Інформаційна технологія для моделювання української мови жестів / Ю.Г. Кривонос, Ю.В. Крак, О.В. Бармак [та ін.] // Штучний інтелект. – 2009. – № 3. – С. 186-197.
2. Кульбіда С.В. Українська дактилологія : науково-методичний посібник / Кульбіда С.В. – К. : Педагогічна думка, 2007. – 256 с.
3. Hamburg Notation System:HamNoSys. – Режим доступу : Web site: http://www.sign-lang.uni-hamburg.de/projects/hamnosys.html

4. SignWriting Site. – Режим доступу : Web site: http://www.signwriting.org/
5. William C. Stokoe, Jr. Sign language structure: An outline of the visual communication systems of the american deaf / William C. Stokoe, Jr. – University of Buffalo, 1960. – 91 p.
6. Hennessy J.L. Computer Architecture, A Quantitative Approach / J.L. Hennessy and D.A. Patterson. – [2nd Edition]. – San Francisco (California) : Morgan Kaufmann Publishers, Inc., 1996.

Literatura

1. Krivonos Ju.G. №3 Shtuchnyj intelekt. 2009. S. 186-197.
2. Kul'bida S.V. Ukraïns'ka daktilologija: Naukovo-metodichnij posibnik. K: Pedagogichna dumka. 2007. 256 s.
3. Hamburg Notation System:HamNoSys.http://www.sign-lang.uni-hamburg.de/projects/hamnosys.html
4. SignWriting Site. Web http://www.signwriting.org/
5. William C. Stokoe, Jr. Sign language structure: An outline of the visual communication systems of the american deaf. University of Buffalo. 1960. 91 p.
6. Hennessy J.L. Computer Architecture, A Quantitative Approach. 2nd Edition. Morgan Kaufmann Publishers. Inc. San Francisco, California. 1996.

Ю.В. Крак, Ю.В. Барчукова, Б.А. Троценко

Построение моделей дактилем для синтеза дактильной информации

В статье рассматривается подход к построению спецификаций движений человека, а именно дактильного жестового языка. Рассматривается информационная модель руки человека. Описан процесс построения спецификации для дактильного жестового языка. Проведен сравнительный анализ семи жестовых языков. Предложен подход и программная реализация на платформе .Net системы передачи асинхронных сообщений, применяемых для проведения параллельных вычислений на всех ядрах процессора, избежания блокирования основного потока программы, написания сценариев деятельности программы.

Yu.V. Krak, Yu.V. Barchukova, B.A. Trotsenko

Creation of Models of Dactyl Units for the Synthesis of Dactyl Information

An approach for creating a specification of finger-spelling process is suggested in the article. The specification is based on a suggested informational model of a human palm. The process of creating a specification is described in detail. Seven sign languages have been analyzed. For the proper and fast software implementation of finger-spelling synthesis an approach of an asynchronous messaging is suggested and implemented within .Net framework. The approach allows using all available CPU cores for computing hand states, prevents UI thread from blocking, uses scenario-driven approach to software development.

Стаття надійшла до редакції 16.06.2011.